14/06/2019

Adeel Asghar

Mr. Adnerson - SDD

User Manual

SDD – HSC Assessment Task 3

Included in this document is a description of how my program functions to allow a user to create a cloze passage form either an entered piece of text or an imported text file stored on their device. How to save the cloze passage and word bank or complete the cloze passage within the program will also be described in this document; the user manual.

Upon opening the program, the user is presented with the Home page where they click a **Start button** to begin the game or access the first Help page to see how to play. Once the program is started the user can either import or type text into the provided textbox before entering a value to substitute the *n*th variable to remove every *n*th word then clicking on the **Remove button** to edit the entered text and create a cloze passage with numbered blank spaces and a word bank of all removed words. Prior to removing words, the user can clear the text within the textbox and start over, go back to the Home page or access the Help page, however, once the **Remove button** is clicked the user cannot clear or edit the text.

Once the user has created their cloze passage they click the **CREATE button** to move on to the next form/step which presents them with a choice. The user can either complete their cloze passage within the program or save the cloze passage and word bank into a location within their device.

IF the user clicks on the **Save button** then they are taken to the Print form which will display their cloze passage and word bank within textboxes whilst disabling editing to either. The user can print their cloze passage into their device onto a selected file location or they can end the program or go back to the Completion page to complete the cloze passage instead.

IF the user chooses to click the **Complete Now button** then they’re taken to the DoNow2 form where they are presented with 3 textboxes. The first 2 from the left display the cloze passage and word bank whilst the third textbox is reserved to allow the user to complete their cloze passage within the program. The user can click on the **Help button** to access the second Help2 form where they can read how to complete their cloze passage using what is displayed to them. Once the user clicks the **Back button** form the Help2 page they then return to the DoNow2 page to complete their cloze passage by entering a word from the text box and their assumed space number derived form the other 2 textboxes.

Once the user is satisfied and has completed their cloze passage, they click on the **FINISH button** to move them onto the Finish form which, once they click on the **END button**, they end the program and finish the game.

**Online Help**

I have implemented two help forms within my program which provides the user with instant help on how to use my program to create and complete/print a cloze passage. Once the user clicks on the help buttons located on the Home, Creation and DoNow form they will be taken to a help form to understand what to do to create and complete their cloze passage. The DoNow form has a unique Help form directed to/from it which instruct the user in how to complete the cloze passage, whilst the Home and Creation form have another Help page which directs how to create their cloze passage within the program using the functions displayed.

**Algorithm**

BEGIN Home

IF btnStart = clicked THEN

Display Creation form

ENDIF

IF btnHelp = clicked THEN

Display Help form

Variable = 0

ENDIF

END Home

BEGIN Creation

IF btnBack = clicked THEN

Display Home form

ENDIF

IF btnHelp = clicked THEN

Display Help form

Variable = 1

ENDIF

WHILE btnImport = clicked

Open file

Read each line of file

Print lines of file into rtbenter

Close file

ENDWHILE

Bankarray() = rtbenter entered text

Count <= temparray.length – 1

IF count = nth = 0 THEN

Text = text + temparray(count)

Temparray(count) = str(space) + “\_\_\_\_\_”

Space = Space + 1

ENDIF

Count = count + 1

WHILE count <= temparray.length – 1

Text = text + “ “ + temparray(count)

Count = count + 1

ENDWHILE

IF btnCREATE = clicked THEN  
 display Completion form

ENDIF

END Creation

BEGIN Help

btnGotIt! = clicked

IF variable = 0 THEN

Display Home form

ELSE

Display Creation form

ENDIF

END Help

BEGIN Completion

IF btnnow = clicked THEN

Display DoNow2 form

ENDIF

IF btnSave = clicked Then

Display Print form

ENDIF

END Completion

BEGIN DoNow2

IF btnBack = clicked THEN

Display Completion form

ENDIF

WHILE valu <= Creation.bankarray.length – 1

Rtbtext.text = rtbtext.text & “ “ & Creation.bankarray(valu)

valu = value + 1  
ENDWHILE

IF btnHelp = clicked THEN

Display Help2 form

ENDIF

IF btnFINISH = clicked THEN

Display messagebox

Display Finish form

ENDIF

END DoNow2

BEGIN Help2

IF btnGotIt1! = clicked THEN

Display DoNow2 form

ENDIF

END Help2

BEGIN Print

IF btnBack = clicked THEN

Display Completion form

ENDIF

btnPrint = clicked

WHILE Print.copies = 1

Print rtb1 & Creation.bankarray(valu)

ENDWHILE

WHILE valu <= Creation.bankarray.length – 1

Rtb1 = rtb1 & “ “ & Creation.bankarray(valu)

valu = valu + 1

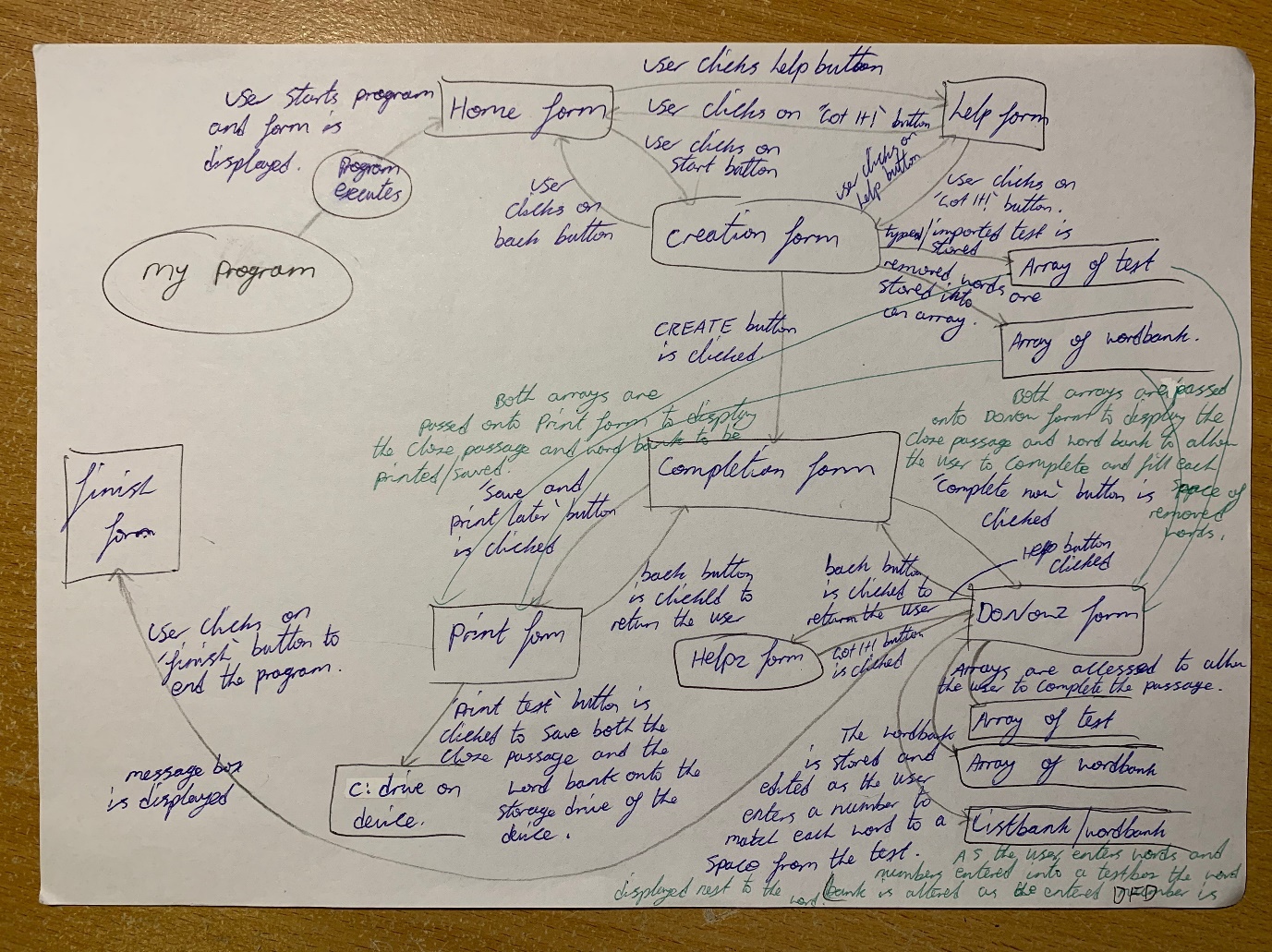
ENDWHILE

END Print

**Data Dictionary**

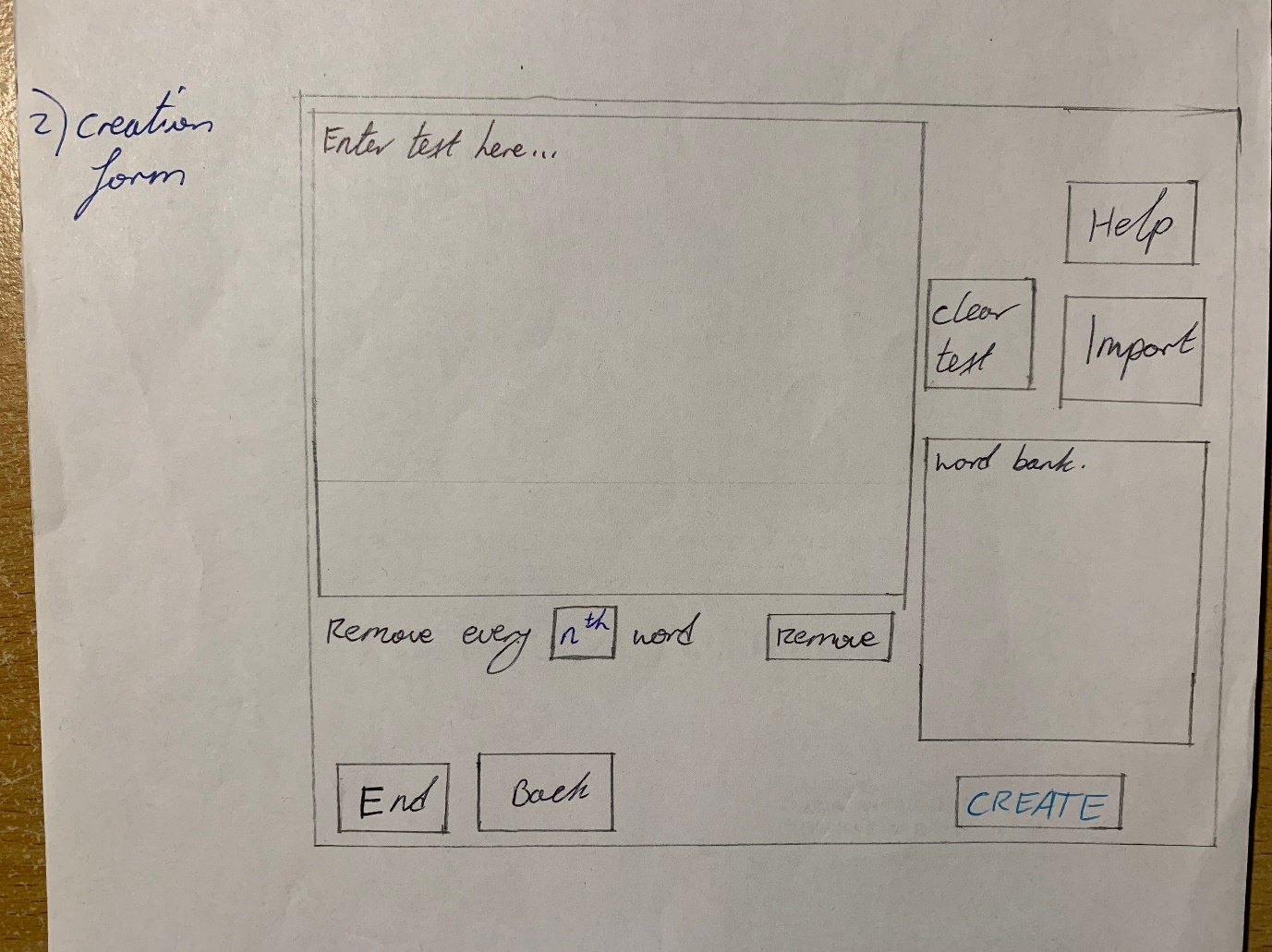
|  |  |  |  |
| --- | --- | --- | --- |
| Variable Name | Data Type | Description | Example |
| Variable | Boolean | Determines which page the back button returns to from the Help form | Variable = 0/1 |
| Count | Integer | Word counter | Lblcount.text = count - 1 |
| bankarray() | String | Stores the removed words into the array | Every nth word that is removed from the text |
| Check | Boolean | Checks to see if bankarray is written to | If check = False Then |
| Bank | String | Stores the word bank as a variable | Me.bank = rtbbank.text |
| Cloze | String | Stores the text of the cloze passage | Me.cloze = rtbenter.text |
| Path | String | Declares the variable for reading a file | Path = OpenFile.FileName.ToString() |
| FileNum | Integer |  |  |
| temparray() | String | Variable to store bankarray | Dim temparray As String() = bankarray |
| nth | Integer | The numerical value of the text entered ‘nth textbox’ | Dim nth As Integer = Txtnth.Text |
| Count (within a private sub) | Integer | Variable equalling the length of temparray | While count <= temparray.Length – 1 |
| text | String | The text replacing words within the text/cloze passage | text = text + vbcrlf + temparray(Count) |
| space | Integer | Array variable declaring the word being removed | Temparray(count) = str(space) + “\_\_\_\_” |
| Valu | Integer | Variable equalling the length of bankarray declared within Creation form | While valu <= Creation.bankarray.Length – 1 |
| Strarray | String | Array storing the cloze passage | Strarray(index) = words(index) |
| Index | Integer | Variable equalling the length of the words array | While index <= Words.Length – 1 |
| Words | String() | Variable equalling the array that stores the removed words; word bank. | Dim Words As String() = Creation.bank.Split(vbCr) |
| Count | Integer | Counts how many words are stored within the word bank | While count <= LBbank.ItemHeight |
| Strarray | String | Is the variable that makes count equal an array | If strarray(count) = txtword.text Then |

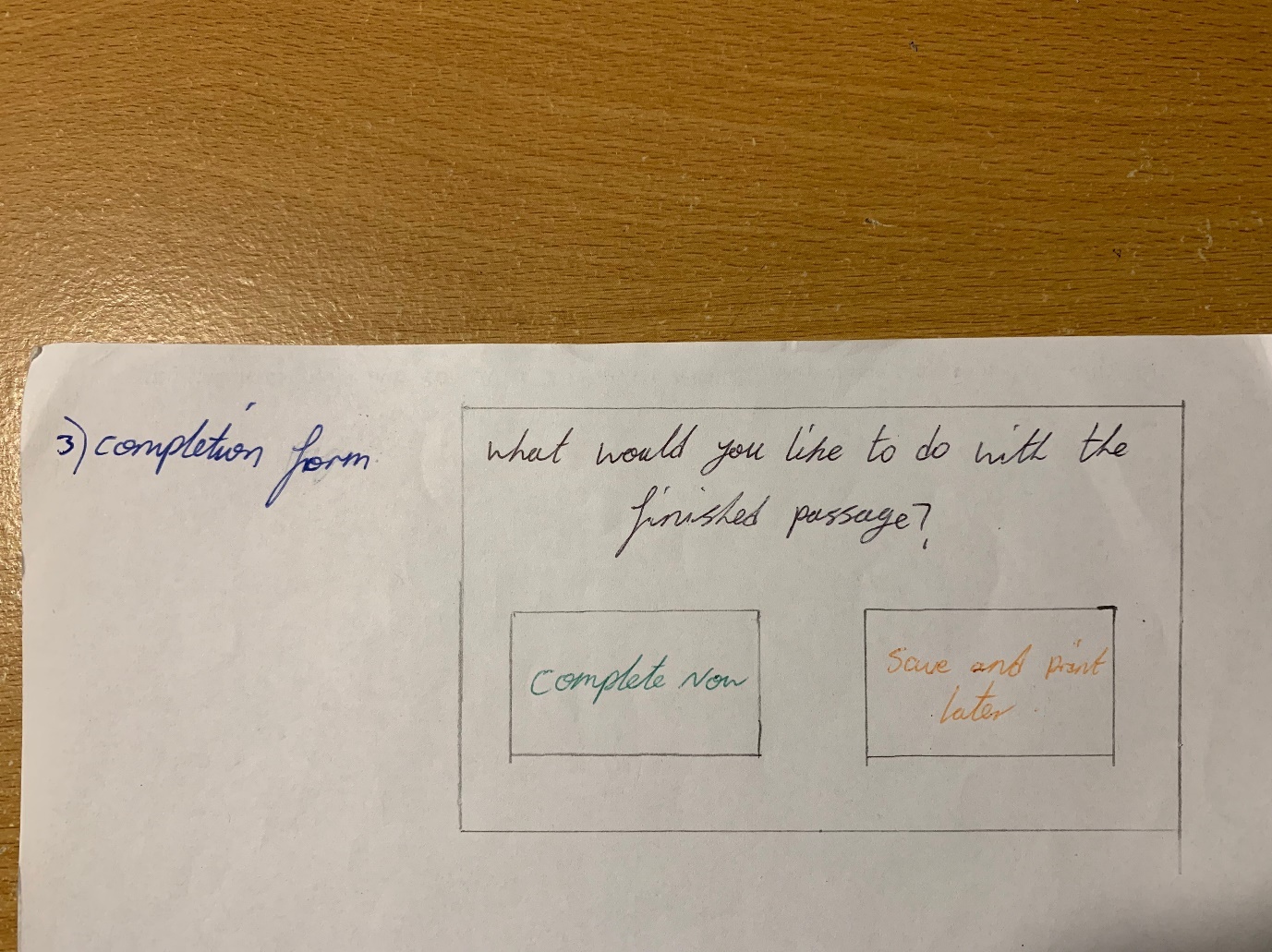
**Data-flow diagram**

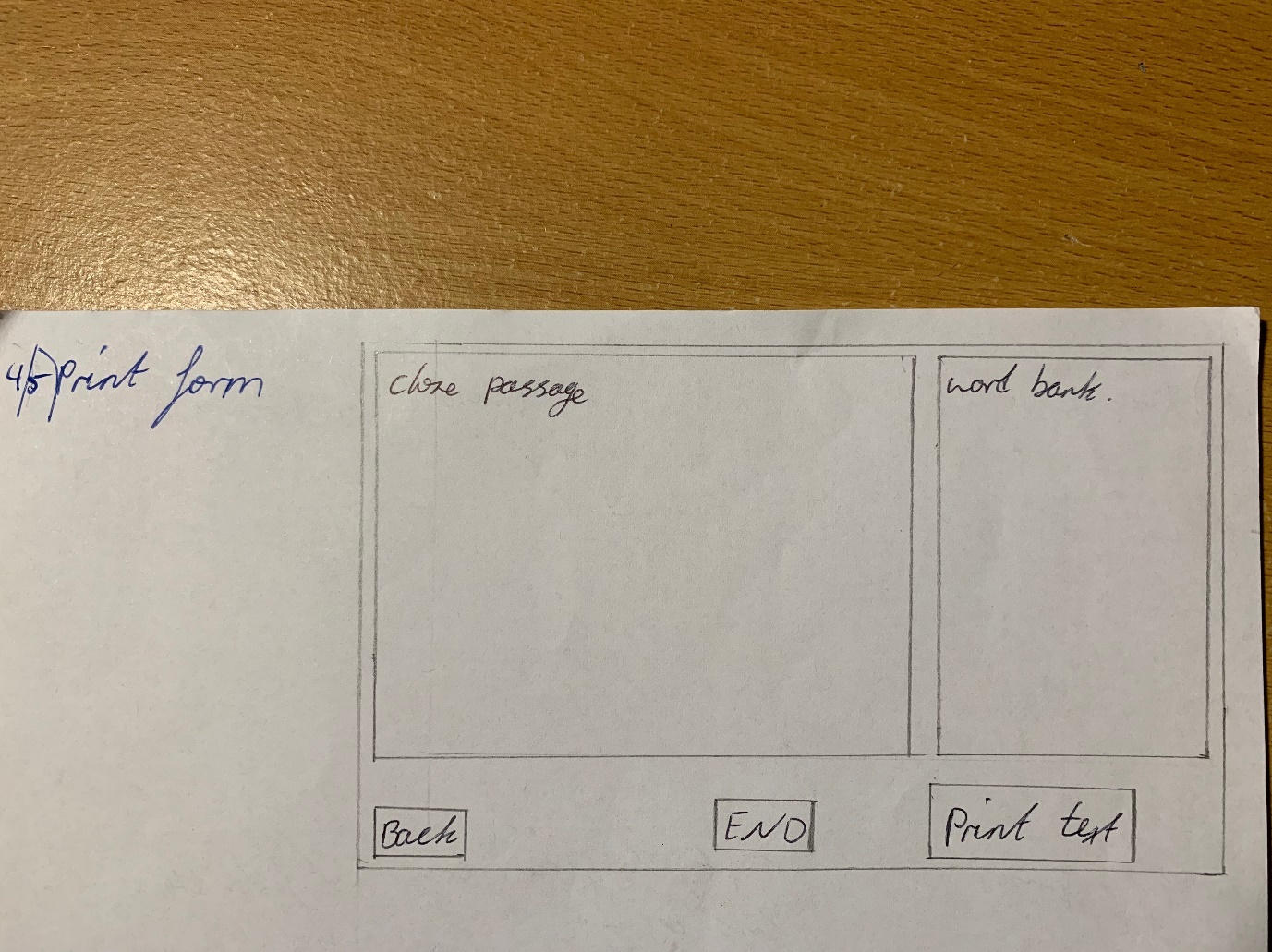
****

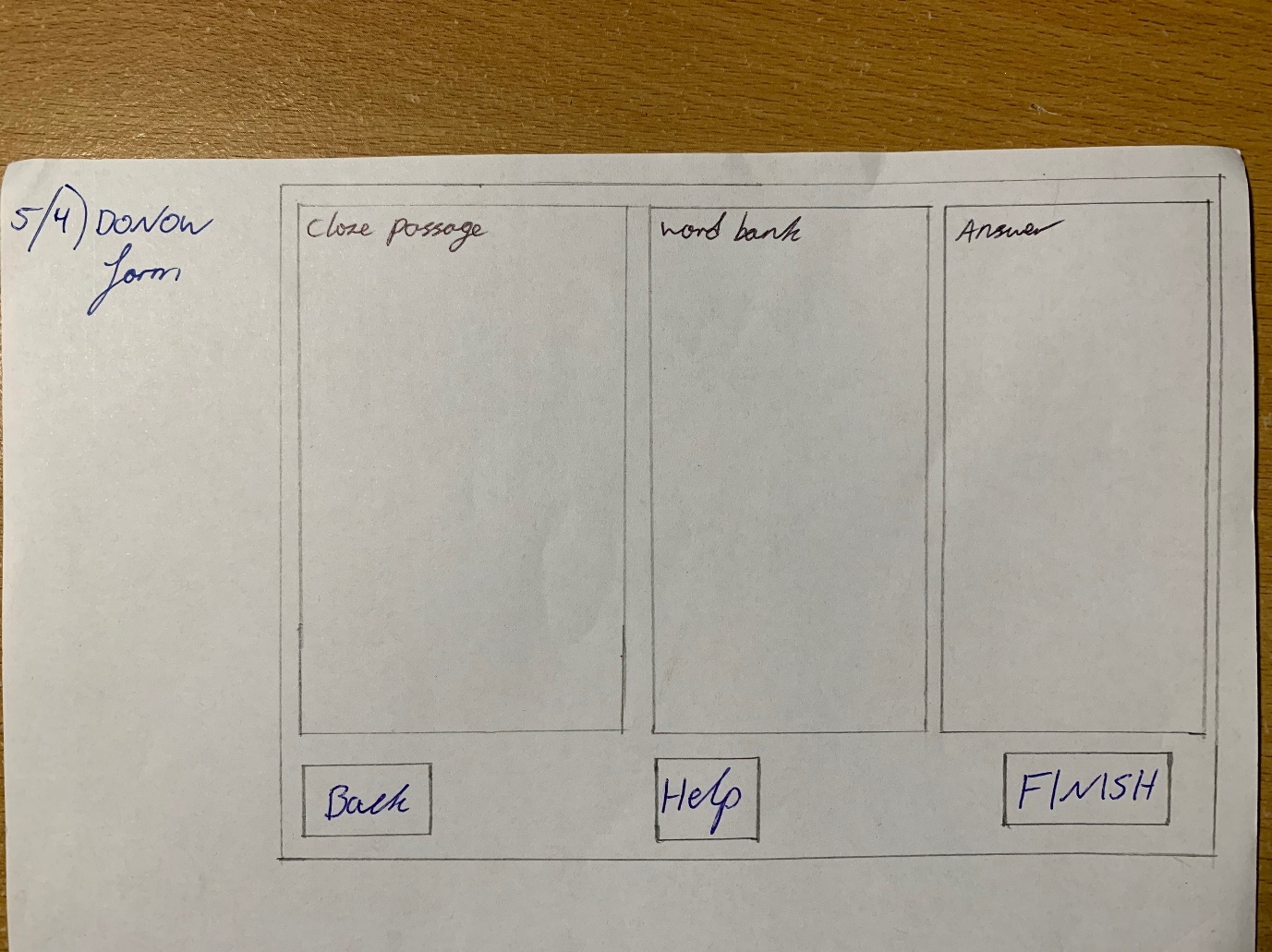
**Screen layout diagram**

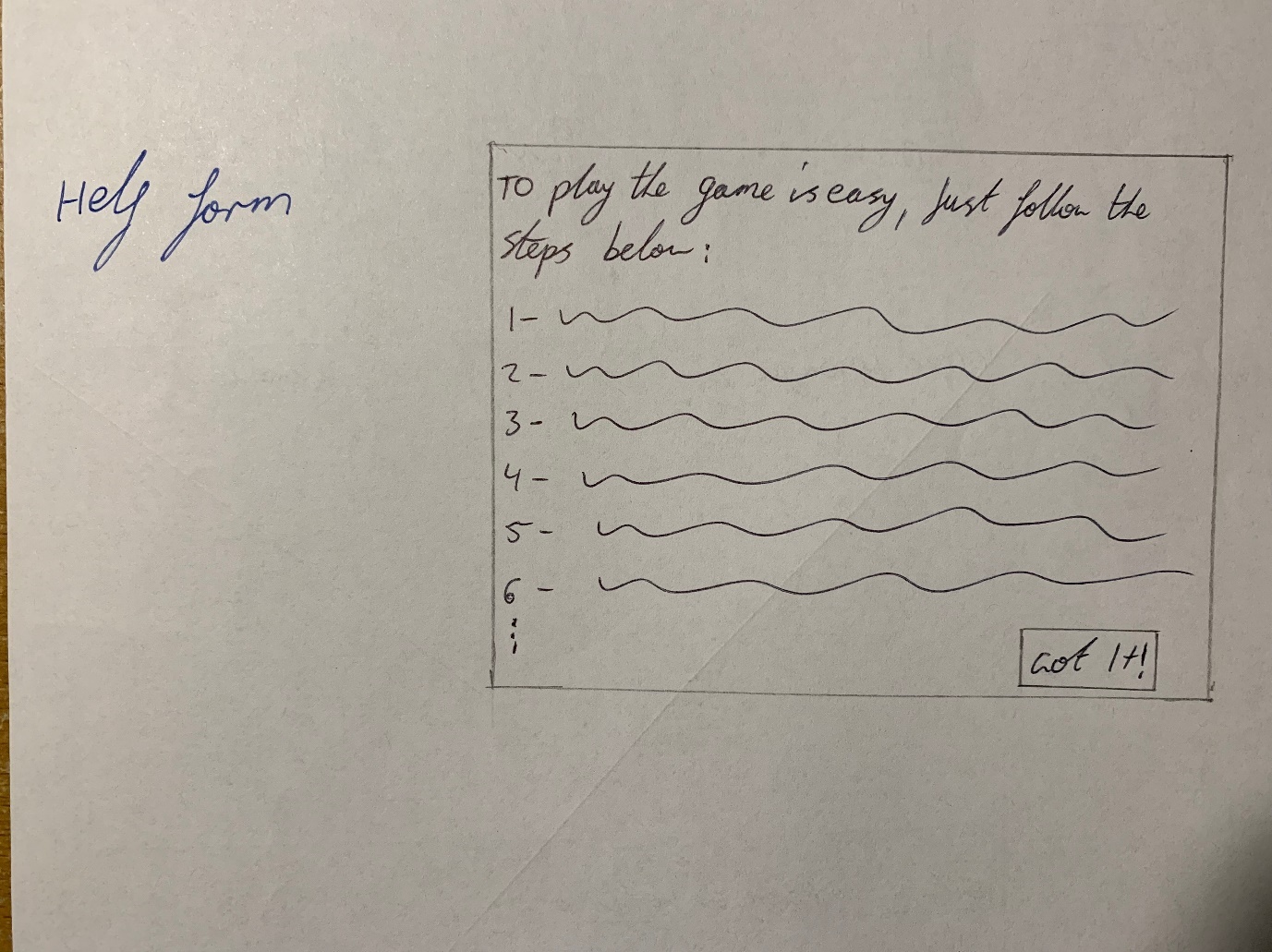
****

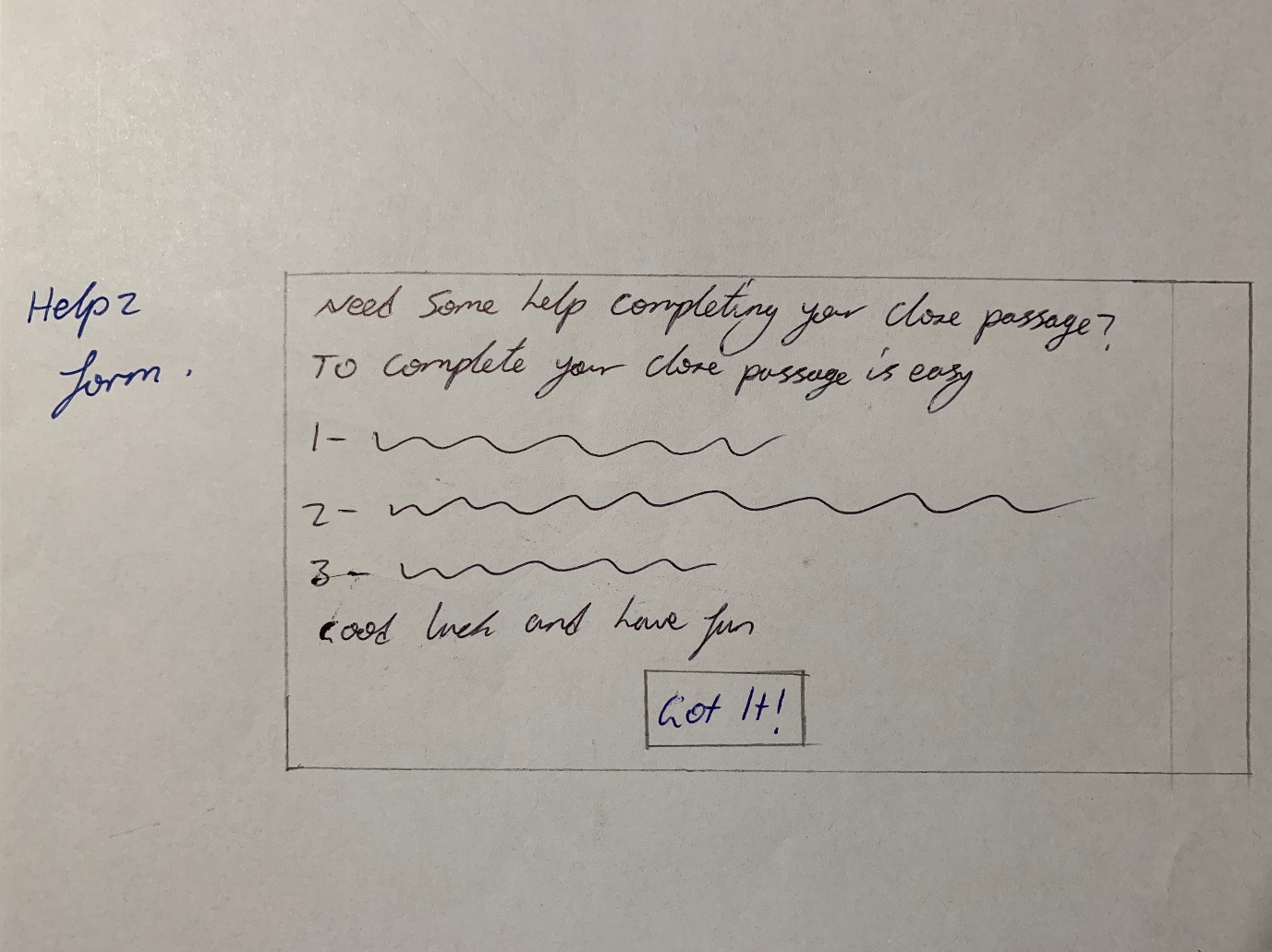
****

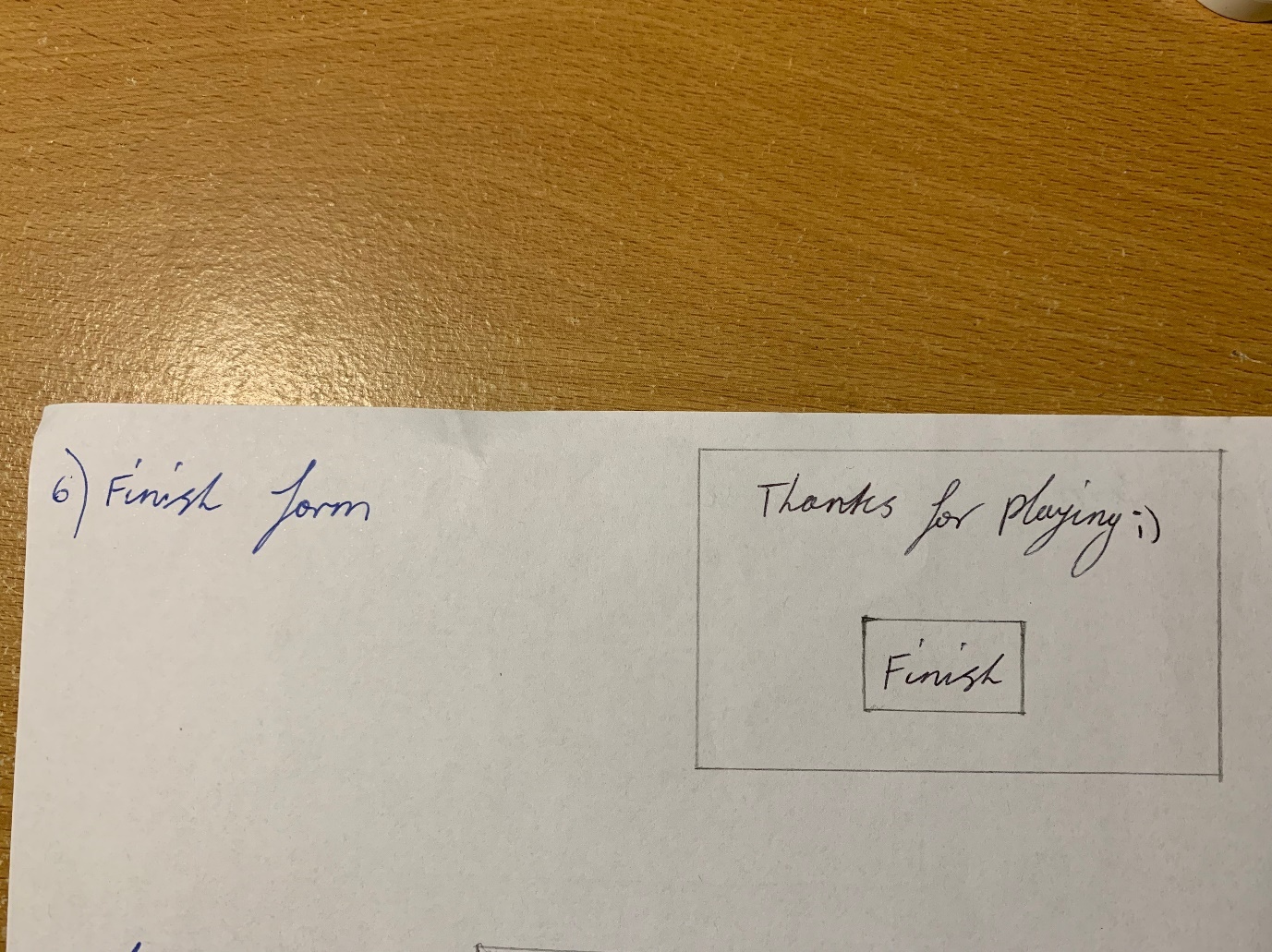
****

****

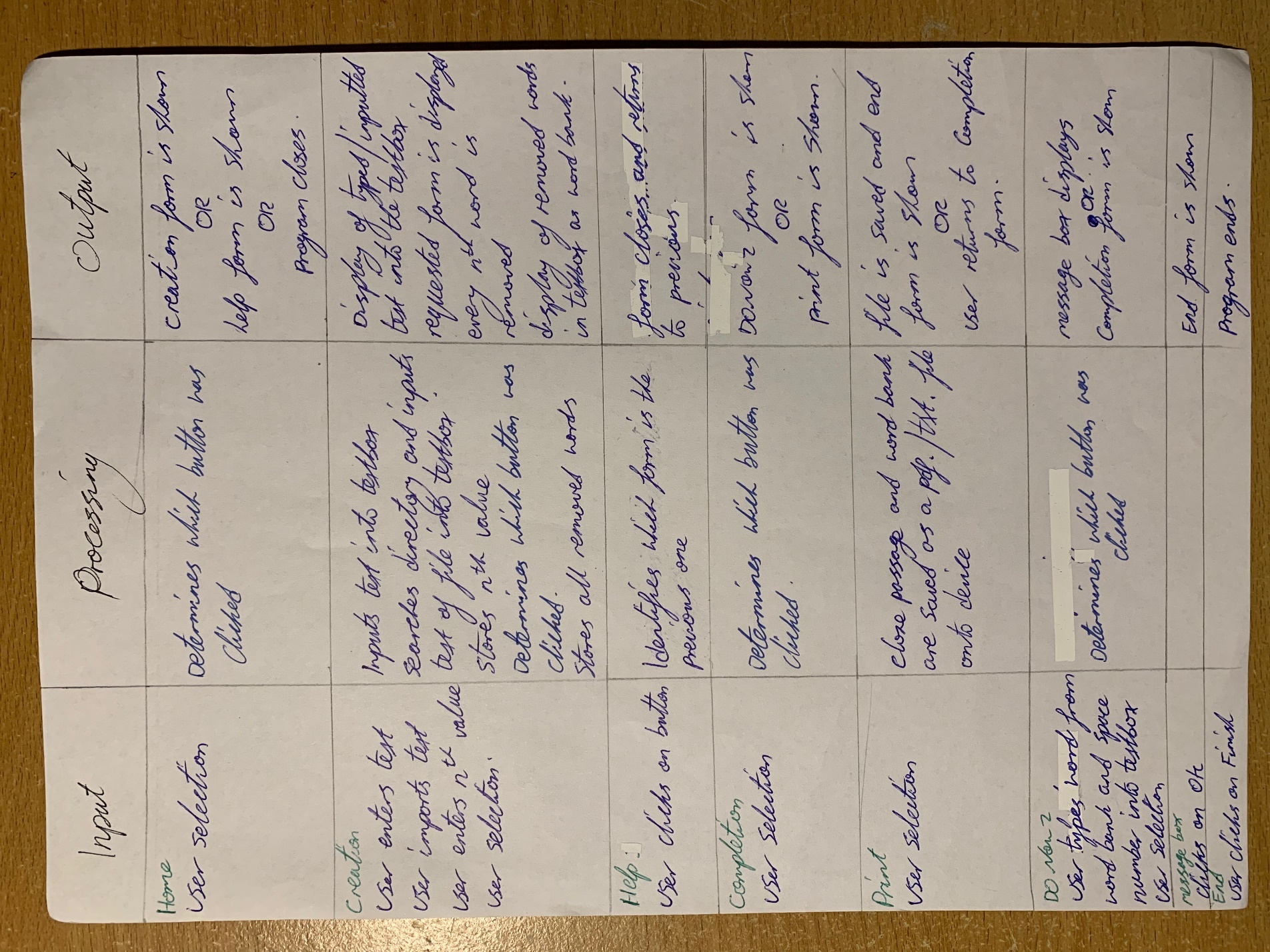
****

****

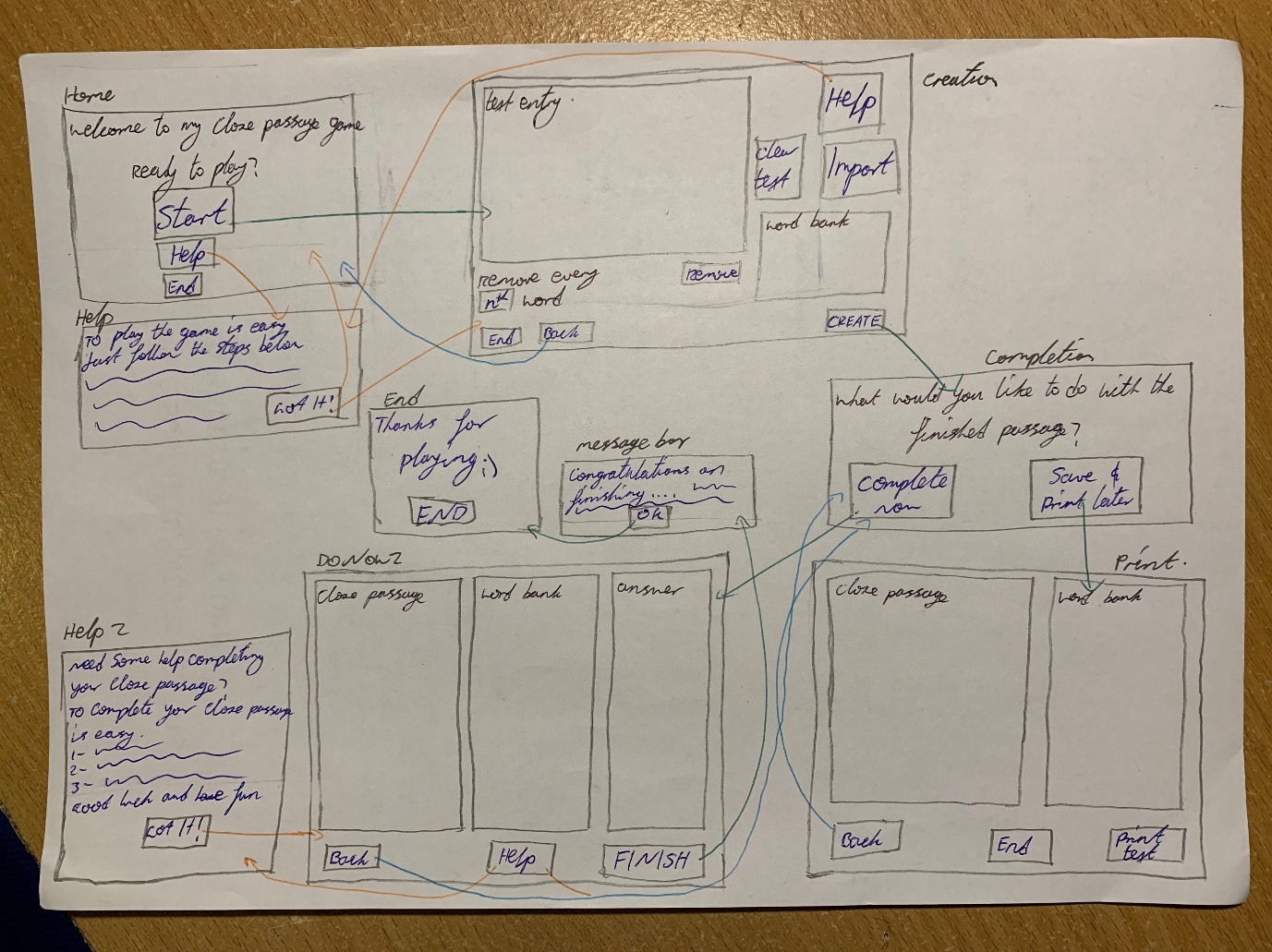
****

****

**IPO**

****

**Storyboard**

****

**Data types and structures used and why**

I have incorporated numerous data types and structures into my program for numerous reasons ranging from requiring that specific data type to enabling special functions and allowing for a more user friendly and logical program. The different data types that I have used include:

* String; the text entered into the textboxes by the user and the program use string data types to store the data.
* Boolean; the data used to determine which form the back button returns to within the Help form as either the Home or Creation form.
* Integer; the word counter within the Creation form and the indexes within the arrays are all stored as an integer data type.

The different structures I have implemented include:

* Split; this allows the text entered/imported into the textbox to be used to create the cloze passage within the program to be stored into an array where each word is saved into each index and can be called on to remove specific words which are multiples of the entered nth value.

**User interface**

I have ensured to present an intuitive and clear display for the user to follow as they progress through my program. I have included text within each textbox to indicate what must be entered and made buttons and their labels clear for easy navigation. All elements of each form have been made as clear as possible of its function as to indicate its necessity for appropriate use of the program and its position within the form. Each element within each form has been designed to optimise user friendliness and ease of use throughout the progress of my program from start to finish.

**Elements on each screen and justification**

Within each screen there are directory buttons and textboxes that allow the user to progress through my program and create a cloze passage from within the program itself. I have designed large buttons with clear simple text to make it easy to identify what each button does and where each can be found. This is to allow for ease of use as well as user-friendliness. Below is a list of the different elements I have used and, what they do within my program:

* Textboxes; allows the user to enter the *n*th value to remove the desired words from their entered body of text.
* Scroll bars; displays a scroll bar within rich textboxes when the text displayed is too long and allows the user to scroll up or down to view the text within
* Command/Buttons; placed throughout my program to provide navigation to other forms and perform functions such as saving the cloze passage and word bank, removal of words, importing of text and the clearing of text from rich textboxes
* Rich textboxes; used in numerous forms to allow text to be entered to imported, to display the cloze passage and word bank and allow for completion